

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日                      2 0 0 3 年    3 月 2 4 日  
Date of Application:

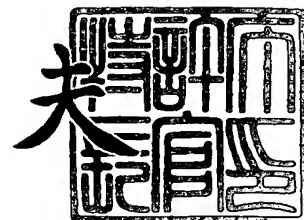
出 願 番 号                      特 願 2 0 0 3 - 0 8 0 1 6 8  
Application Number:  
[ST. 10/C]:                      [ J P 2 0 0 3 - 0 8 0 1 6 8 ]

出      願      人                      株式会社東京三菱銀行  
Applicant(s):                      日本アイ・ビー・エム株式会社

2 0 0 3 年    8 月 2 5 日

特許庁長官  
Commissioner,  
Japan Patent Office

今 井 康 夫



【書類名】 特許願

【整理番号】 JP9030027

【提出日】 平成15年 3月24日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F157:00

【発明者】

    【住所又は居所】 東京都千代田区丸の内 2 丁目 7 番 1 号 株式会社東京三  
    菱銀行内

    【氏名】 高橋 一興

【発明者】

    【住所又は居所】 千葉県千葉市中瀬 1 番地 1 日本アイ・ビー・エム株式  
    会社 幕張事業所内

    【氏名】 長井 泉

【発明者】

    【住所又は居所】 千葉県千葉市中瀬 1 番地 1 日本アイ・ビー・エム株式  
    会社 幕張事業所内

    【氏名】 星島 洋一

【特許出願人】

    【識別番号】 598049322

    【氏名又は名称】 株式会社 東京三菱銀行

【特許出願人】

    【識別番号】 592073101

    【氏名又は名称】 日本アイ・ビー・エム株式会社

【代理人】

    【識別番号】 100104880

    【弁理士】

    【氏名又は名称】 古部 次郎

【選任した代理人】

【識別番号】 100118201

【弁理士】

【氏名又は名称】 千田 武

【手数料の表示】

【予納台帳番号】 081504

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データベース更新処理システム、データベース更新のための更新データ入力方法、更新データ処理方法、およびプログラム

【特許請求の範囲】

【請求項 1】 データベースを更新する複数の更新データをブロックとして保管するデータキューと、

前記データキューに格納される複数の前記ブロックを関連付けてパッケージ単位とし、当該パッケージの内容を記述したパッケージ情報を保管する制御キューと、

前記制御キューから前記パッケージ情報を読み取り、前記パッケージにより関連付けられている前記ブロックを前記データキューから取り出してデータベース更新処理を行う処理タスクと

を含むデータベース更新処理システム。

【請求項 2】 前記処理タスクを複数設け、

前記複数の処理タスクは、前記制御キューを共用して並列処理を行うことを特徴とする請求項 1 記載のデータベース更新処理システム。

【請求項 3】 前記制御キューは、前記データベース更新処理の順序を守る必要があるデータに関しては、一つの前記パッケージ内に収納されるようにパッケージングがなされることを特徴とする請求項 1 記載のデータベース更新処理システム。

【請求項 4】 データベース更新業務の障害時に、当該データベース更新業務で処理していたブロックに関するパッケージ情報を一時保留する保留キューを更に含む請求項 1 記載のデータベース更新処理システム。

【請求項 5】 データベースに対する更新処理を行うためのデータベース更新処理システムであって、

ソート済み更新データファイルから更新データを読み込み、所定の更新データ単位に独立したバッファを用いてブロッキングを行う手段と、

メッセージ ID と、複数の更新データを含むブロックをデータキューに書き出す手段と、

所定数のブロックが前記データキューに書き出された後に、当該ブロックの処理単位を特定するパッケージ情報を制御キューに書き出す手段とを含むデータベース更新処理システム。

【請求項 6】 前記制御キューに書き込まれる前記パッケージ情報は、前記データキューに書き出された前記ブロックに含まれる前記メッセージ ID と同一のメッセージ ID によって、当該データキューと関連付けがなされていることを特徴とする請求項 5 記載のデータベース更新処理システム。

【請求項 7】 データベースに対する更新処理を行うためのデータベース更新処理システムであって、

更新データのブロックを複数まとめたパッケージ情報を保有する制御キューから当該パッケージ情報を取得するパッケージ情報取得手段と、

前記制御キューに保有される前記パッケージ情報から、前記ブロックを保有するデータキューを特定し、特定された当該データキューから当該ブロックを取得するブロック取得手段と、

前記データベースに対して更新処理を行う更新業務部に、前記ブロック取得手段により取得された前記ブロックを出力する出力手段と

を含むデータベース更新処理システム。

【請求項 8】 前記出力手段は、前記ブロック取得手段により取得された前記ブロックに含まれる更新データの内容ごとに決められたデータベース更新業務を呼び出して、当該ブロックを当該データベース更新業務に提供することを特徴とする請求項 7 記載のデータベース更新処理システム。

【請求項 9】 前記制御キューは、複数の前記パッケージ情報を保有すると共に、

前記ブロック取得手段は、特定の前記パッケージ情報に対して関連付けされたデータキューから前記ブロックを取得することを特徴とする請求項 7 記載のデータベース更新処理システム。

【請求項 10】 前記更新業務部に障害があった場合に、障害が発生したデータベース更新業務で処理していたブロックに対する前記パッケージ情報を保留キューに出力する保留キュー出力手段を更に含む請求項 7 記載のデータベース

更新処理システム。

【請求項 1 1】 データベースに対する更新データをソート済み更新データファイルから読み込むステップと、

読み込まれた前記更新データから、所定の更新データ単位にてブロッキングを行うステップと、

前記ブロッキングにより形成されたブロックをデータキューに書き出すステップと、

所定数のブロックが前記データキューに書き出された後に、当該ブロックをまとめて処理するためのパッケージ情報を制御キューに書き出すステップと

を含むデータベース更新のための更新データ入力方法。

【請求項 1 2】 前記制御キューには複数の前記パッケージ情報が書き込まれると共に、書き込まれた複数の前記パッケージ情報の中の特定のパッケージ情報は、複数の前記ブロックが保管される前記データキューと関連付けがなされていることを特徴とする請求項 1 1 記載のデータベース更新のための更新データ入力方法。

【請求項 1 3】 更新データのブロックを複数まとめたパッケージ情報を保有する制御キューから当該パッケージ情報を取得するステップと、

前記制御キューに保有される前記パッケージ情報から、前記ブロックを保有するデータキューを特定するステップと、

特定された前記データキューから前記ブロックを取得するステップと、

データベースに対して更新処理を行う更新業務部に、取得された前記ブロックを出力するステップと

を含むデータベース更新のための更新データ処理方法。

【請求項 1 4】 前記更新業務部にて障害があった場合に、障害が発生したデータベース更新業務で処理していた前記ブロックに対する前記パッケージ情報を保留キューに出力するステップを更に含む請求項 1 3 記載のデータベース更新のための更新データ処理方法。

【請求項 1 5】 コンピュータに、

データベースに対する更新データをソート済み更新データファイルから読み込

む機能と、

読み込まれた前記更新データから、所定の更新データ単位にてブロッキングを行う機能と、

前記ブロッキングにより形成されたブロックをデータキューに書き出す機能と

、  
所定数のブロックが前記データキューに書き出された後に、当該ブロックをまとめて処理するためのパッケージ情報を制御キューに書き出す機能と

を実現させるプログラム。

【請求項 16】 コンピュータに、

更新データのブロックを複数まとめたパッケージ情報を保有する制御キューから当該パッケージ情報を取得する機能と、

前記制御キューに保有される前記パッケージ情報から、前記ブロックを保有するデータキューを特定する機能と、

特定された前記データキューから前記ブロックを取得する機能と、

データベースに対して更新処理を行う更新業務部に、取得された前記ブロックを出力する機能と

を実現させるプログラム。

【請求項 17】 コンピュータに

前記更新業務部にて障害があった場合に、障害が発生したデータベース更新業務で処理していた前記ブロックに対する前記パッケージ情報を保留キューに出力する機能を更に実現させる請求項 16 記載のプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、限られた時間内に大量の更新データを効率良く、かつ、高速に処理するデータベース更新処理システム等に関する。

【0002】

【従来の技術】

近年、コンピュータと端末装置あるいはコンピュータ同士が通信回線を介して

接続されるオンラインシステムでは、ユーザから送信されたメッセージに応答して、所定のデータ処理が施された後、所定の結果を通知するトランザクション処理が広く実施されている。従来のトランザクション処理システムでは、システム内に一つまたは複数のキューを持ち、そのキュー内に蓄えられたトランザクションデータを、シングルタスクまたはマルチタスクで、順次または独立して処理する構成が採用されている。

### 【0 0 0 3】

シングルタスクでトランザクションデータを順次に処理する従来のシステムでは、単位時間あたりにおけるトランザクションデータ処理件数の向上には限界があった。そのために、マルチタスクによる並列処理が検討されてきたが、一つのキューに蓄積された多数トランザクションデータをマルチタスクで効率良く処理することは非常に難しい。そこで、独立に処理可能な複数のトランザクションデータを蓄積するキューと、このキューの制御情報を含む制御テーブルと、キューに蓄積されたトランザクションデータに対して同じ処理を行う複数のタスクを稼動する手段とを備えることで、複数タスクの効率的な並列処理により処理速度を向上することができる並列トランザクション処理システムが提案されている(例えば、特許文献1参照。)

### 【0 0 0 4】

#### 【特許文献1】

特開平11-53202号公報(第3-4頁、図1)

### 【0 0 0 5】

#### 【発明が解決しようとする課題】

ここで、例えば、金融機関の情報システムにおけるデータベースを例に挙げると、これらの情報システムでは、(i)一日に数千万件以上の大量データを取り扱う。(ii)例えば勘定系のログ以外の他業務システムから渡されるファイルにも対応する。(iii)多種多様なデータベース更新を行う。(iv)データ量の増加に柔軟に対応する。といった数々の複雑な条件が、データベースの構築に際して前提とされている。

### 【0 0 0 6】



例えば、情報系システムでのデータベースの構築では、これらの条件のもとで所定のデータベース更新処理を行うことが必要となる。しかしながら、データは一定時間間隔で大量に受け渡されるために、限られた時間内にデータベース更新処理を完了させる仕組みが必要となる。また、データ処理中にデータベースを更新するプログラムなどで障害が発生した場合には、障害データを回避させ、後続のデータを処理することが必要となる。上記特許文献1の技術だけでは、例えば、上記金融機関の情報システムが有する各種条件に対して充分に対処することができない。

#### 【0007】

本発明は、以上のような技術的課題を解決するためになされたものであって、その目的とするところは、例えば制御キューという新たなコンポーネント、およびデータのブロッキングとパッケージングを採用することで、大量のデータを高速に処理する並列処理の実現と共に、データベースのコンテンションを回避することにある。

また他の目的は、例えば個々のパッケージ単位で並列処理を行うことで、データベース更新の処理効率を向上させることにある。

更に他の目的は、例えばパッケージングする更新データのキーを事前にソートすることで、データベースのコンテンション(端末同士の使用権の奪い合い)を極小化することにある。

また更に他の目的は、例えばプログラム障害などの起因となったデータを含むパッケージ単位にデータ処理を保留することで、データ量が大量であった場合でも障害時に適切に対処することにある。

#### 【0008】

##### 【課題を解決するための手段】

かかる目的のもと、本発明では、データベースを更新する  $n$  個の更新データを一つのブロックとして物理的にブロッキングし、この複数ブロックを一つのまとまりとして、論理的にパッケージングした。そして、このパッケージ単位にパッケージ情報を作成し、制御キューに格納(保管)している。制御キューに格納(保管)されたパッケージ情報は、データ処理部(ストリーマ)によって読み込まれ、

読み込んだデータ処理部(ストリーマ)がパッケージ情報に対応するパッケージを処理するように構成した。これによって、複数パッケージ間での並列処理が可能になると共に、一つのパッケージ内のブロックを順次処理することが可能となる。

#### 【0009】

即ち、本発明が適用されるデータベース更新処理システムは、データベースを更新する複数の更新データをブロックとして保管するデータキューと、このデータキューに格納される複数のブロックを関連付けてパッケージ単位とし、このパッケージの内容を記述したパッケージ情報を保管する制御キューと、制御キューからパッケージ情報を読み取り、パッケージにより関連付けられているブロックをデータキューから取り出してデータベース更新処理を行う処理タスクとを含む。

#### 【0010】

ここで、この処理タスクを複数設け、複数の処理タスクは、制御キューを共用して並列処理を行うことを特徴とすれば、データベースのコンテンションを回避し、処理効率を向上させることができる点で好ましい。また、この制御キューは、データベース更新処理の順序を守る必要があるデータに関しては、一つのパッケージ内に収納されるようにパッケージングがなされることを特徴とすれば、連続した処理を行う場合などに、順序処理を可能とする点で優れている。

#### 【0011】

また、データベース更新業務の障害時に、データベース更新業務で処理していたブロックに関するパッケージ情報を一時保留する保留キューを更に含むことを特徴とすれば、障害があった更新データをデータキューより移動させることなく、処理をバイパスさせることができる点で好ましい。

#### 【0012】

一方、本発明は、データベースに対する更新処理を行うためのデータベース更新処理システムであって、ソート済み更新データファイルから更新データを読み込み、所定の更新データ単位に独立したバッファを用いてブロッキングを行う手段と、メッセージIDと複数の更新データを含むブロックをデータキューに書き

出す手段と、所定数のブロックがデータキューに書き出された後に、ブロックの処理単位を特定するパッケージ情報を制御キューに書き出す手段とを含む。

#### 【0013】

ここで、この制御キューに書き込まれるパッケージ情報は、データキューに書き出されたブロックに含まれるメッセージIDと同一のメッセージIDによって、データキューと関連付けがなされていることを特徴とすることができる。

#### 【0014】

他方、本発明が適用されるデータベース更新処理システムは、パッケージ情報取得手段により、更新データのブロックを複数まとめたパッケージ情報を保有する制御キューからパッケージ情報を取得し、ブロック取得手段により、制御キューに保有されるパッケージ情報から、ブロックを保有するデータキューを特定し、特定されたデータキューからブロックを取得する。また、出力手段により、データベースに対して更新処理を行う更新業務部に、ブロック取得手段により取得されたブロックを出力している。

#### 【0015】

ここで、この出力手段は、ブロック取得手段により取得されたブロックに含まれる更新データの内容ごとに決められたデータベース更新業務を呼び出して、ブロックをデータベース更新業務に提供することを特徴としている。尚、この制御キューは、複数のパッケージ情報を保有すると共に、ブロック取得手段は、特定のパッケージ情報に対して関連付けされたデータキューからブロックを取得することを特徴としている。更に、この更新業務部にて障害があった場合に、障害が発生したデータベース更新業務で処理していたブロックに対するパッケージ情報を保留キューに出力する保留キュー出力手段を含むことを特徴とすれば、処理タスクでの再処理を自動的に防止することが可能となる点で優れている。

#### 【0016】

他の観点から捉えると、本発明が適用されるデータベース更新のための更新データ入力方法は、データベースに対する更新データをソート済み更新データファイルから読み込むステップと、読み込まれた更新データから、所定の更新データ単位にてブロッキングを行うステップと、ブロッキングにより形成されたブロッ

クをデータキューに書き出すステップと、所定数のブロックがデータキューに書き出された後に、ブロックをまとめて処理するためのパッケージ情報を制御キューに書き出すステップとを含む。ここで、この制御キューには複数のパッケージ情報が書き込まれると共に、書き込まれた複数のパッケージ情報の中の特定のパッケージ情報は、複数のブロックが保管されるデータキューと関連付けがなされていることを特徴とすることができる。

#### 【0017】

また、他の観点から捉えると、本発明が適用されるデータベース更新のための更新データ処理方法は、更新データのブロックを複数まとめたパッケージ情報を保有する制御キューからパッケージ情報を取得するステップと、制御キューに保有されるパッケージ情報から、ブロックを保有するデータキューを特定するステップと、特定されたデータキューからブロックを取得するステップと、データベースに対して更新処理を行う更新業務部に、取得されたブロックを出力するステップと、更新業務部にて障害があった場合に、障害が発生したデータベース更新業務で処理していたブロックに対するパッケージ情報を保留キューに出力するステップとを含む。

#### 【0018】

ここで、本発明は、コンピュータに所定の機能を実現させるプログラムとして把握することができる。即ち、本発明が適用されるプログラムは、コンピュータに、データベースに対する更新データをソート済み更新データファイルから読み込む機能と、読み込まれた更新データから、所定の更新データ単位にてブロッキングを行う機能と、ブロッキングにより形成されたブロックをデータキューに書き出す機能と、所定数のブロックがデータキューに書き出された後に、ブロックをまとめて処理するためのパッケージ情報を制御キューに書き出す機能とを実現させる。

#### 【0019】

また、本発明が適用されるプログラムは、コンピュータに更新データのブロックを複数まとめたパッケージ情報を保有する制御キューからパッケージ情報を取得する機能と、制御キューに保有されるパッケージ情報から、ブロックを保有す

るデータキューを特定する機能と、特定されたデータキューからブロックを取得する機能と、データベースに対して更新処理を行う更新業務部に、取得されたブロックを出力する機能と、更新業務部にて障害があった場合に、障害が発生したデータベース更新業務で処理していたブロックに対するパッケージ情報を保留キューに出力する機能とを実現させる。

#### 【0020】

尚、これらのプログラムとしては、コンピュータを顧客に対して提供する際に、装置の中にインストールされた状態にて提供される場合の他、コンピュータに実行させるプログラムをコンピュータが読取可能に記憶した記憶媒体にて提供する形態が考えられる。この記憶媒体としては、例えばCD-ROM媒体や磁気テープ等が該当し、CD-ROM読取装置や磁気テープ装置等によってプログラムが読み取られて実行される。また、これらのプログラムは、例えば、プログラム伝送装置によってネットワークを介して提供される形態がある。このプログラム伝送装置は、例えば、ホスト側のサーバに設けられ、プログラムを格納するメモリと、ネットワークを介してプログラムを提供するプログラム伝送手段とを備えている。

#### 【0021】

##### 【発明の実施の形態】

以下、添付図面に示す実施の形態に基づいて本発明を詳細に説明する。

図9は、本実施の形態において使用可能であるコンピュータシステムの代表的なハードウェア構成の一例を示したものである。コンピュータシステム100は、中央処理装置(CPU)101とメインメモリ104を含んでいる。CPU101とメインメモリ104は、バス102を介して、補助記憶装置としてのハードディスク装置113、130と接続されている。また、フレキシブルディスク装置120、MO装置128、CD-ROM装置126、129などのリムーバブルストレージ(記録メディアを交換可能な外部記憶システム)が関連するフレキシブルディスクコントローラ119、IDEコントローラ125、SCSIコントローラ127などを介してバス102へ接続されている。

#### 【0022】

フレキシブルディスク装置 120、MO装置 128、CD-ROM装置 126、129などのリムーバブルストレージには、それぞれフレキシブルディスク、MO、CD-ROMなどの記憶媒体が挿入され、このフレキシブルディスク等やハードディスク装置 113、130、ROM 114には、オペレーティングシステム(OS)と協働して中央処理装置(CPU) 101等に命令を与え、本実施の形態を実現するためのコンピュータプログラムのコードを記録することができる。メインメモリ 104にロードされることによってコンピュータプログラムは実行される。コンピュータプログラムは圧縮し、また複数に分割して複数の媒体に記録することもできる。

#### 【0023】

コンピュータシステム 100は、更に、ユーザインターフェイスハードウェアとして、マウス等のポインティングデバイス 107、キーボード 106や視覚データをユーザに提示するためのディスプレイ 112を有することができる。ポインティングデバイス 107やキーボード 106は、キーボードマウスコントローラ 105によって制御される。また、ディスプレイ 112は、グラフィックス制御チップ(VGA) 108によりVRAM 109を用いてグラフィックス展開された画像が表示される。更に、パラレルポート 116を介してプリンタ(図示せず)と接続することや、シリアルポート 115を介してモデム(図示せず)を接続することが可能である。また、シリアルポート 115及びモデムを介し、また、通信アダプタ 118(イーサネット(米ゼロックス社登録商標)カードやトークンリングカード)等を介してネットワークに接続し、他のコンピュータ等と通信を行うことが可能である。

#### 【0024】

スピーカ 123は、オーディオコントローラ 121によってD/A変換(デジタル/アナログ変換)された音声信号をアンプ 122を介して受け取り、音声として出力する。また、オーディオコントローラ 121は、マイクロフォン 124から受け取った音声情報をA/D変換(アナログ/デジタル変換)し、システム外部の音声情報をシステムに取り込むことを可能にしている。

#### 【0025】

以上の説明により、本発明の実施の形態におけるコンピュータシステム 100 は、メインフレーム、ワークステーション、通常のパーソナルコンピュータ(PC)、ラップトップPC、ノートブックPC、パームトップPC、ネットワークコンピュータ等の情報処理装置、または、これらの組み合わせによって実現されることが容易に理解されるであろう。ただし、これらの構成要素は例示であり、そのすべての構成要素が本実施の形態における必須構成要素となるわけではない。

#### 【0026】

本実施の形態に使用されるコンピュータシステム 100 の各ハードウェア構成要素を、複数のマシンを組み合わせ、それらに機能を配分し実施する等の種々の変更は、当業者によって容易に想定され得るものであり、それらの変更は、当然に本発明の思想に包含される概念である。

#### 【0027】

本発明は、ハードウェア、ソフトウェア、またはハードウェア及びソフトウェアの組み合わせとして実現可能である。ハードウェアとソフトウェアの組み合わせによる実行において、所定のプログラムを有するコンピュータシステム 100 における実行が典型的な例として挙げられる。かかる場合、所定のプログラムがコンピュータシステム 100 にロードされ実行されることにより、所定のプログラムは、コンピュータシステム 100 を制御し、本発明にかかる処理を実行させる。このプログラムは、任意の言語・コード・表記によって表現可能な命令群から構成される。そのような命令群は、システムが特定の機能を直接、または 1. 他の言語・コード・表記への変換、2. 他の媒体への複製、のいずれか一方もしくは双方が行われた後に、実行することを可能にするものである。もちろん、本発明は、そのようなプログラム自体のみならず、プログラムを記録した媒体もその範囲に含むものである。本発明の機能を実行するためのプログラムは、フロッピーディスク、MO、CD-ROM、DVD、ハードディスク装置 113、130、ROM 114、MRAM、RAM 等の任意のコンピュータ読み取り可能な記録媒体に格納することができる。かかるプログラムは、記録媒体への格納のために、通信回線で接続する他のコンピュータシステムからダウンロードしたり、他

の記録媒体から複製したりすることができる。また、かかるプログラムは、圧縮し、または複数に分割して、単一または複数の記録媒体に格納することもできる。

#### 【0028】

本実施の形態のシステムにおいては、図9の各機能ブロックは図1に示すハードウェア構成を持つコンピュータシステムのハードウェア資源と各ハードウェア資源において実行されるオペレーティングシステム、ミドルウェア、アプリケーションプログラムなどのソフトウェアが協働することによって実現される。

#### 【0029】

次に、本実施の形態における特徴的な構成について詳述する。

図1は、本実施の形態におけるデータベース更新処理システムの全体構成を示した図である。図1に示すデータベース更新処理システムは、ログ選別処理部11、更新データファイル12、ソート部13、ソート済み更新データファイル14、および入力処理部30を備えている。ログ選別処理部11は、記帳系からのトランザクションログ(トランザクション処理にて各種DB等を更新した記録)を入力してログを分割し、ログ選別処理を施している。更新データファイル12では、記帳系での処理順に更新データが蓄積されている。ソート部13では、更新データファイル12の内容が主キー順およびDB(データベース)更新通番により並び替えられる。ソート済み更新データファイル14は、ソート部13により主キー順およびDB更新通番にて昇順ソートされた順次ファイルである。また、入力処理部30では、ソート済み更新データファイル14に格納されているデータに基づいて、ログ読み込み・振り分け処理が行われる。

#### 【0030】

また、データベース更新処理システムは、本実施の形態における特徴的な構成として、データキュー21、制御キュー22、保留キュー23、およびデータ処理部(ストリーマ)50を備えている。データキュー21は、更新データをn個まとめたブロック(更新データメッセージ)を保管(蓄積)するキューである。制御キュー22は、複数のブロック(更新データメッセージ)をまとめたパッケージ情報(パッケージ数、データキュー21の名前など)を保管(蓄積)するキューである。



保留キュー 23 は、データ処理部 50 にてプログラム障害などが発生した場合に、処理中に退避させる制御メッセージを保管(蓄積)するキューである。処理タスクであるデータ処理部(ストリーマ) 50 は、データキュー 21 より更新データメッセージを取得してデータベース更新処理を実行している。

#### 【0031】

更に、データベース更新処理システムは、データ処理部 50 から更新データメッセージを受け取る更新業務プログラムからなる更新業務部 20 を有し、この更新業務部 20 は、更新処理振分部 24、DB 更新プログラムである DB 更新部 25 を有している。また、更新業務部 20 による更新処理対象である更新対象 DB 26 を備えている。この更新処理振分部 24 は、データ処理部 50 から渡されたブロックの内容を確認し、内容ごとに決められている DB 更新部 25 にブロックを渡している。また、DB 更新部 25 は、複数の DB 更新業務プログラムによって構成されており、受け取ったブロックをデブロック(ブロック解除)し、業務処理の必要なものには処理を施し、対応する各種更新対象 DB 26 に対して更新を実行している。

#### 【0032】

ここで、本実施の形態におけるパッケージとブロッキングについて説明する。

図 2(a), (b) は、複数のデータ処理部(ストリーマ) 50 が同一キーを持ってデータベースの更新を行う場合の並列処理の例を示した図である。図 2(a) に示すように、同一キー(ここでは、CIF # 1)を持つ複数レコードを、データ処理部(ストリーマ) 50 でそのまま並列処理しようとする、複数のデータ処理部(ストリーマ) 50 が同一キーを持ってデータベースの更新を行うことから、遅延が発生し、結果的には並列処理ができなくなる。そこで、解決策の一つとして、図 2(b) のようにデータをブロッキングする方法が考えられる。しかしながら、例えば、ブロック長を数十 K バイト以内で構築する等、ブロック長には限界があることから、更新するものを限定することができない。ここで、同一キーが最大ブロック長を超えて続く場合について説明する。例えば、最大ブロック長が 30 件で、同一キー(ここでは、CIF # 1)のものが 100 件あった場合を考える。この場合は、この 100 件を全てまとめて 1 つのブロックにすることができない

。かかる場合には、例えば、同一キーを有する 4 個のブロック (3 0 件×3 ブロック + 1 0 件×1 ブロック) が作成される。そのために、同一キーを持つ 4 個のブロックをデータ処理部 (ストリーマ) 5 0 で並列処理すると、データベースのコンテンションが生じるおそれがある。即ち、ブロックだけの観点では、図 2 (a) と同様に、遅延が発生し、結果的に並列処理ができない状況が生じ得る。

#### 【0 0 3 3】

そこで、本実施の形態では、図 3 に示すように、ブロックをパッケージ化 (グループ化) することで、各データ処理部 (ストリーマ) 5 0 は、異なったキーを持つデータを更新することが可能となり、遅延の発生を防止することができる。ここで、「パッケージ」とは、データ処理部 5 0 がデータ順序処理を保証する単位である。例えば、同一キーのものが 1 0 0 件あった場合に、ブロック化することのできる、例えば、3 0 件のブロックを 3 つ作り、残り 1 0 件で 1 つのブロックを作り、これらを 1 つの単位としてパッケージングする。このように、並列処理が可能な論理的な単位であるパッケージという概念を用い、並列処理を実行している。

#### 【0 0 3 4】

次に、これらの特徴的な機能を実現するために、各構成要素にて実行される各処理について詳述する。

図 4 は、入力処理部 3 0 にて実行される処理を示したフローチャートである。入力処理部 3 0 の出力は、データキュー 2 1 および制御キュー 2 2 に書き出されるが、大量データを効率良く処理するために、入力側の処理を「メインタスク」、出力側の処理を「サブタスク」にて実行している。入力処理部 3 0 では、まず、入力側の処理を受け持つ「メインタスク」が開始され、ソート済み更新データファイル 1 4 が「メインタスク」にて読み込まれる (ステップ 1 0 1)。ここで、入力ファイルが終了したか否かが判断される (ステップ 1 0 2)。入力ファイルが終了した場合には、後述するステップ 1 0 8 へ移行する。入力ファイルが終了していない場合に、「メインタスク」では、読み込まれた更新データタイプ毎に、出力側の処理を受け持つ「サブタスク」に対して開始を指示する (ステップ 1 0 3)。

## 【0035】

読み込まれた更新データは、更新データタイプが判断されて、該当する更新データを処理する「サブタスク」に渡される(ステップ104)。その後、更新データの処理を行う「サブタスク」では、各々の更新データのタイプ毎に独立したバッファを使用して、更新データのブロッキングが実行される(ステップ105)。「サブタスク」からは、更新データメッセージとして、完成したブロックがデータキューに出力される(ステップ106)。ここで、ブロッキングが実行される「バッファ」は、入力処理部30のアドレススペース内に獲得された記憶域を指している。

## 【0036】

決められた数のブロック(更新データメッセージ)が書き出された後には、それらのブロックと関連付けるためのパッケージ情報として制御メッセージが作成され、「サブタスク」から制御キュー22に書き出されて、同期点が取られる(ステップ107)。この複数ブロックのまとまりがパッケージである。その後、ステップ101に戻り、同様の処理が繰り返される。ステップ102にて、入力ファイルが終了すると、各更新データを処理している「サブタスク」に対して終了が指示される(ステップ108)。更新データを処理する「サブタスク」は、バッファにあるブロックをデータキュー21に書き出し、加えて対応する制御メッセージも制御キュー22に書き出して、同期点を取り(ステップ109)、入力処理が終了する。尚、「同期点」では、1つの作業単位(ある一組の変更が開始されてから、それらが完了するまでの期間)内で、データベースの更新やメッセージングの更新が確定される。

## 【0037】

以上のようにして、入力処理部30から出力されたデータキュー21の更新データメッセージと、制御キュー22の制御メッセージとは、所定の関連付けがなされている。また、制御キュー22およびデータキュー21も、所定の関連付けがなされている。

図5(a),(b)は、メッセージの関連付け、およびキューの関連付けを説明するための図である。図5(a)は、制御キュー22の制御メッセージとデータキュー

ー 21 の更新データメッセージとの関連を示しており、図 5 (b) は、制御キュー 22 とデータキュー 21 との関連を示している。

#### 【0038】

図 5 (a) に示すように、同一パッケージ内の全ての更新データメッセージは、同一のメッセージ ID で関連付けられている。即ち、制御メッセージが有するメッセージ ID と同じメッセージ ID を、同一パッケージ内の全ての更新データメッセージが有している。更新データメッセージには、このメッセージ ID と共に、更新データ数や複数の更新データ等、データベースを更新する情報が格納されている。尚、メッセージ ID は、例えば、入力処理部 30 によって設定される。

#### 【0039】

図 5 (b) では、第 1 ～ 第 3 の 3 つのデータキュー 21 (21-1 ～ 21-3) が制御キュー 22 に関連付けられている。制御キュー 22 とデータキュー 21 とは、 $1:n$  ( $n \geq 1$ ) の関係にある。ここで同一パッケージの全ての更新データメッセージは、同一のデータキュー 21 に収納される。

#### 【0040】

このようにして、複数のブロックデータは、論理的に関連付けられてパッケージ単位とされ、その内容が記述された制御メッセージが作成される。そして、制御キュー 22 にこの制御メッセージが保管されている。また、データベースを更新するブロックデータは、更新データメッセージとしてデータキュー 21 に保管される。

#### 【0041】

次に、データ処理部 (ストリーマ) 50 について説明する。

図 6 は、データ処理部 (ストリーマ) 50 の機能構成を説明するためのブロック図である。処理タスクとして機能するデータ処理部 50 は、全体制御部 51、ストリーム開始部 52、ユーザメッセージ取り出し部 53、ブロック終了部 54、ストリーム終了部 55、およびバックアウト部 56 を備えている。全体制御部 51 では、データ処理部 50 の全体制御が実行される。ストリーム開始部 52 では、起動時にキューのオープンとデータ処理部 50 の制御情報 (ストリーマ制御情報) の初期化が行われる。ユーザメッセージ取り出し部 53 では、制御キュー 2

2 に対して制御メッセージの待受けが行われる。出力手段として機能するブロック終了部 54 は、更新データメッセージの処理が終了したことを確認し、制御キュー 22 に制御メッセージを出力している。また、ストリーム終了部 55 では、データ処理部 50 の終了時にキューのクローズ処理が行われる。更に、バックアウト部 56 は、障害が発生した場合に、該当する制御メッセージを保留キュー 23 に出力している。尚、更新業務部 20 の DB 更新部 25 では、複数の DB 更新業務が実行される。

#### 【0042】

図 7 および図 8 は、主にデータ処理部(ストリーマ) 50 にて実行される処理を示したフローチャートである。図 6 を参照しながら説明すると、ストリーム開始部 52 は、データ処理部(ストリーマ) 50 の起動時に、ストリーマ制御情報域(データ処理部 50 ごとに獲得され管理された記憶領域)に対して初期値化を行い、入力対象のキュー名を設定する(ステップ 201)。ストリーム開始部 52 は、ストリーマ制御情報より入力対象のキュー名を取得し、制御キュー 22、データキュー 21、および保留キュー 23 をオープンする(ステップ 202)。

#### 【0043】

パッケージ情報取得手段およびブロック取得手段の一つとして機能するユーザメッセージ取り出し部 53 は、制御キュー 22 に対して制御メッセージの待受けを行う(ステップ 203)。ここで、制御キュー 22 に対して制御メッセージがあるか否かが判断される(ステップ 204)。制御キュー 22 に制御メッセージがない場合には、所定の時間待機し、その所定時間が経過すると、更新データがない旨を全体制御部 51 へ通知し、全体制御部 51 へ制御を戻す(ステップ 205)。その後、停止指示があるか否かが判断され(ステップ 206)、停止指示がない場合には、ステップ 203 へ戻り、停止指示がある場合には、データ処理部(ストリーマ) 50 の停止時に、制御キュー 22、データキュー 21、および保留キュー 23 のクローズ処理が行われて(ステップ 207)、処理が終了する。

#### 【0044】

ステップ 204 にて、制御キュー 22 に制御メッセージがある場合には、データキュー 21 の名前とパッケージ数、更新データタイプなどからなるパッケージ

情報を取得する(ステップ208)。そして、パッケージ情報より、対応するデータキュー21を特定し、特定されたデータキュー21から更新データメッセージを読み込み(ステップ209)、更新データメッセージを全体制御部51へ渡す。このとき、全体制御部51へは、処理する更新データが存在する旨が知らされる。

#### 【0045】

全体制御部51では、ユーザメッセージ取り出し部53からの情報に基づき、更新データがあるか否かで異なる作業が実行される(ステップ210)。更新データがない場合には、前述したステップ205へ移行する。更新データがある場合には、全体制御部51は、更新処理振分プログラムである更新処理振分部24に更新データメッセージを渡す(ステップ211)。更新処理振分部24では、渡された更新データメッセージの内容が確認され、DB更新部25の中の、更新データの内容ごとに決められたDB更新業務が呼び出されて、更新データメッセージが渡される(ステップ212)。

#### 【0046】

次に、図8を用いて説明すると、DB更新部25の中の、ステップ211で更新データメッセージが渡されたDB更新業務では、渡された更新データメッセージをもとに業務処理が行われ、更新対象DB26に対してDBの更新が実行される(ステップ213)。DB更新業務では、DB更新の実行結果の状況が確認され、リターンコードが更新処理振分部24に返される(ステップ214)。更新処理振分部24では、戻されたリターンコードの判定がなされ、必要な後処理が行われて、データ処理部50にリターンコードが返される(ステップ215)。

#### 【0047】

データ処理部50のブロック終了部54は、受け取ったリターンコードが正常である場合には、更新データメッセージの処理が終了したことを確認し、制御キュー22に、制御メッセージを出力する(ステップ216)。その後、データ処理部50における同期点処理として、全体制御部51にて同期点が取られ、更新データメッセージによるデータベース更新処理が確定し(ステップ217)、図7に示すステップ205に戻る。

## 【 0 0 4 8 】

尚、DB 更新部 2 5 の所定の DB 更新業務にて、プログラム障害が発生した場合には、データ処理部 5 0 は、更新した DB および更新データメッセージの戻し時のロールバック処理が行われる。即ち、障害の発生した DB 更新業務にて処理されていた更新データメッセージを処理保留にするために、バックアウト部 5 6 は、該当する制御メッセージを保留キュー 2 3 に出力する。このとき、出力される制御メッセージは、ユーザメッセージ取り出し部 5 3 で取得したオリジナルの制御メッセージが使用される。

## 【 0 0 4 9 】

このように、本実施の形態では、入力処理部 3 0 にて、複数のブロックデータを論理的に関連付けてパッケージ単位とし、その内容を記述した制御メッセージを作成し、制御キュー 2 2 に対して保管するように構成した。また、データベースを更新するブロックデータは、更新データメッセージとして、データキュー 2 1 に保管するように構成した。即ち、本実施の形態では、大量データを高速に処理するための工夫として、並列処理の実現と、データベースコンテンションの回避を行うために、制御キュー 2 2 という新しいコンポーネントおよびデータのブロッキングとパッケージングを提案した。一方、データ処理部 5 0 では、制御キュー 2 2 から制御メッセージを読み取り、関連付けられている更新データメッセージをデータキュー 2 1 から取り出して、DB 更新処理を行わせている。このように、一つの制御キュー 2 2 を複数のデータ処理部 5 0 (ストリーマ) で共用することにより、個々のパッケージ単位でデータ処理部 5 0 (ストリーマ) による並列処理が可能となり、データベース更新処理のパフォーマンス(処理効率)を向上させることができる。また、パッケージングする更新データのキーを事前にソートしておくことにより、データベースのコンテンションを極小化することができる。更に、DB 更新する順序を守る必要があるデータに関しては、一つのパッケージ内に収納されるようにパッケージングすることにより、順次処理が可能となり、特に、銀行系のような連続した入出金データ処理を行う場合に非常に有効である。

## 【 0 0 5 0 】

また、障害時は、データ量が大量なため、プログラム障害時の起因となったデータを含むパッケージ単位にデータ処理が保留される。即ち、DB更新部 2 5 (DB更新業務)の障害時には、パッケージ情報を保有する制御メッセージを一時保留するための保留キュー 2 3 を設け、制御メッセージを退避した。この仕組みによって、障害データをデータキュー 2 1 より移動させることなく処理をバイパスすることが可能となり、後続の別パッケージを処理することが可能となる。これによって、大量処理を行う際に、障害対応を短時間に、かつ自動的に行うことができる。

【 0 0 5 1 】

【発明の効果】

このように、本発明によれば、大量のデータを高速に処理する並列処理の実現と共に、データベースのコンテンションを回避することが可能となる。

【図面の簡単な説明】

【図 1】 本実施の形態におけるデータベース更新処理システムの全体構成を示した図である。

【図 2】 (a), (b)は、複数のデータ処理部(ストリーマ)が同一キーを持ってデータベースの更新を行う場合の並列処理の例を示した図である。

【図 3】 ブロックをパッケージ化(グループ化)する本実施の形態の特徴的な構成を説明するための図である。

【図 4】 入力処理部にて実行される処理を示したフローチャートである。

【図 5】 (a), (b)は、メッセージの関連付け、およびキューの関連付けを説明するための図である。

【図 6】 データ処理部(ストリーマ)の機能構成を説明するためのブロック図である。

【図 7】 主にデータ処理部(ストリーマ)にて実行される処理を示したフローチャートである。

【図 8】 主にデータ処理部(ストリーマ)にて実行される処理を示したフローチャートである。

【図 9】 本実施の形態において使用可能であるコンピュータシステムの代



表的なハードウェア構成の一例を示したものである。

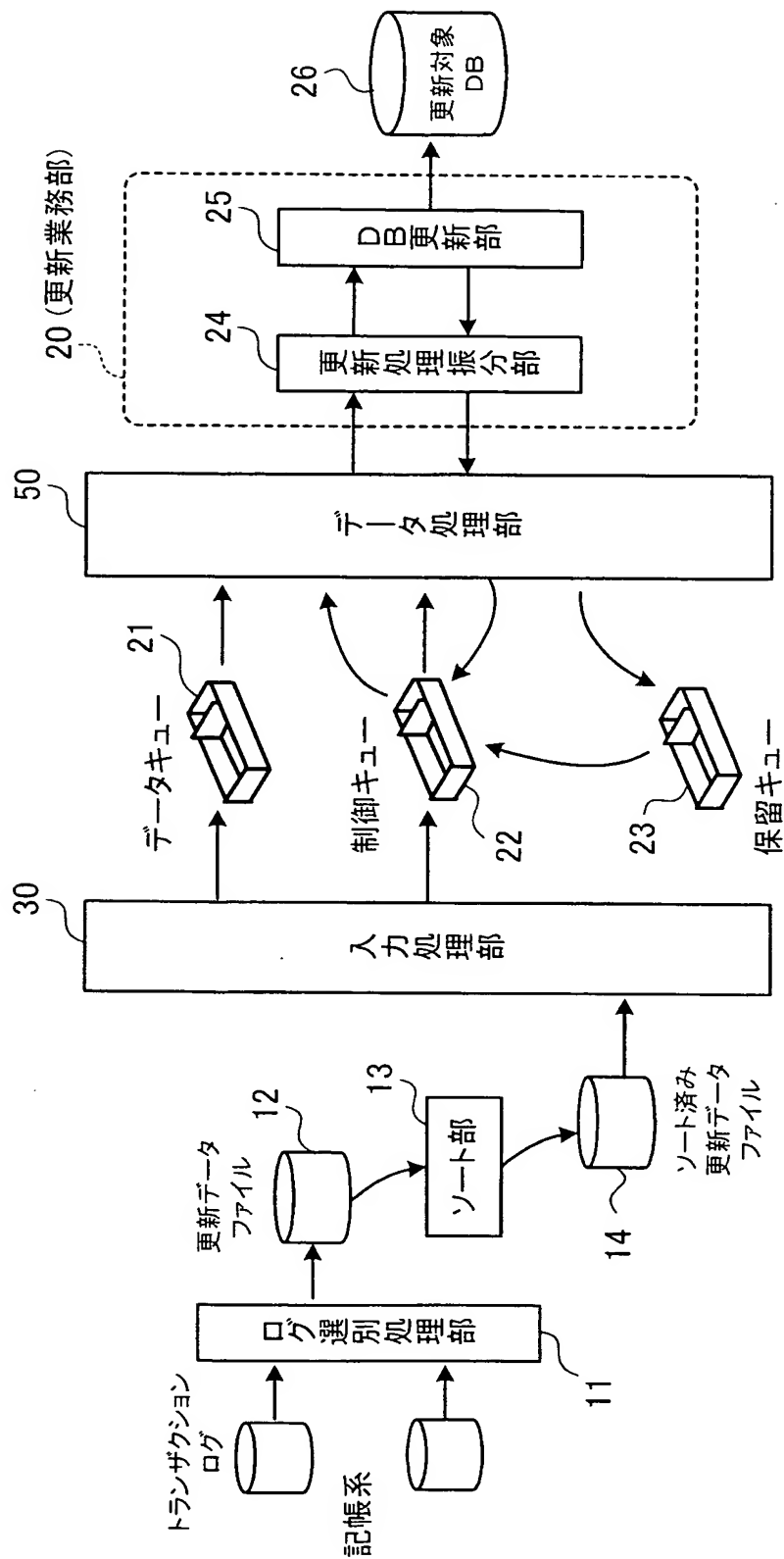
**【符号の説明】**

1 1 …ログ選別処理部、1 2 …更新データファイル、1 3 …ソート部、1 4 …ソート済み更新データファイル、2 0 …更新業務部、2 1 …データキュー、2 2 …制御キュー、2 3 …保留キュー、2 4 …更新処理振分部、2 5 …DB更新部、2 6 …更新対象DB、3 0 …入力処理部、5 0 …データ処理部(ストリーマ、データ処理タスク)、5 1 …全体制御部、5 2 …ストリーム開始部、5 3 …ユーザメッセージ取り出し部、5 4 …ブロック終了部、5 5 …ストリーム終了部、5 6 …バックアウト部

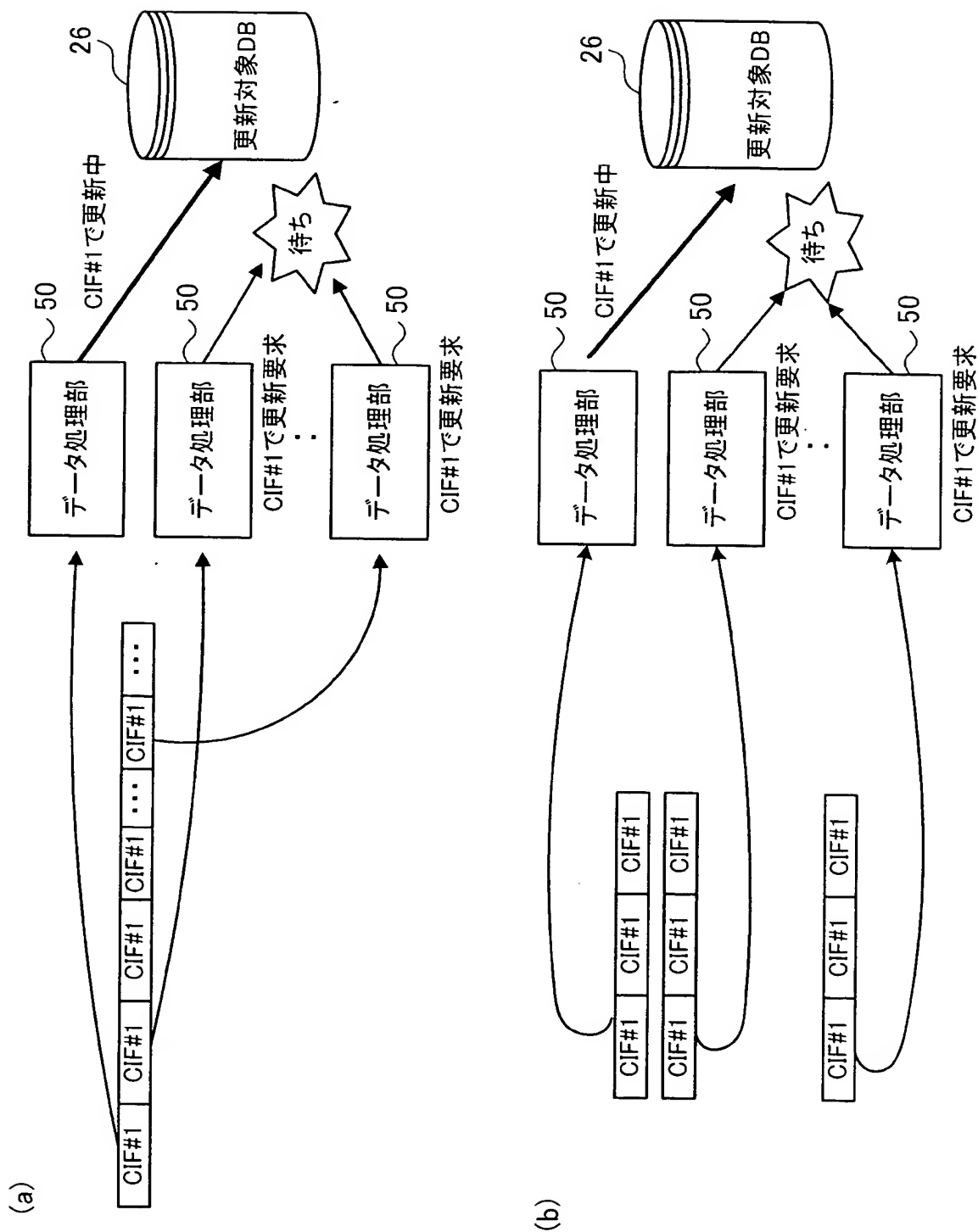
【書類名】

図面

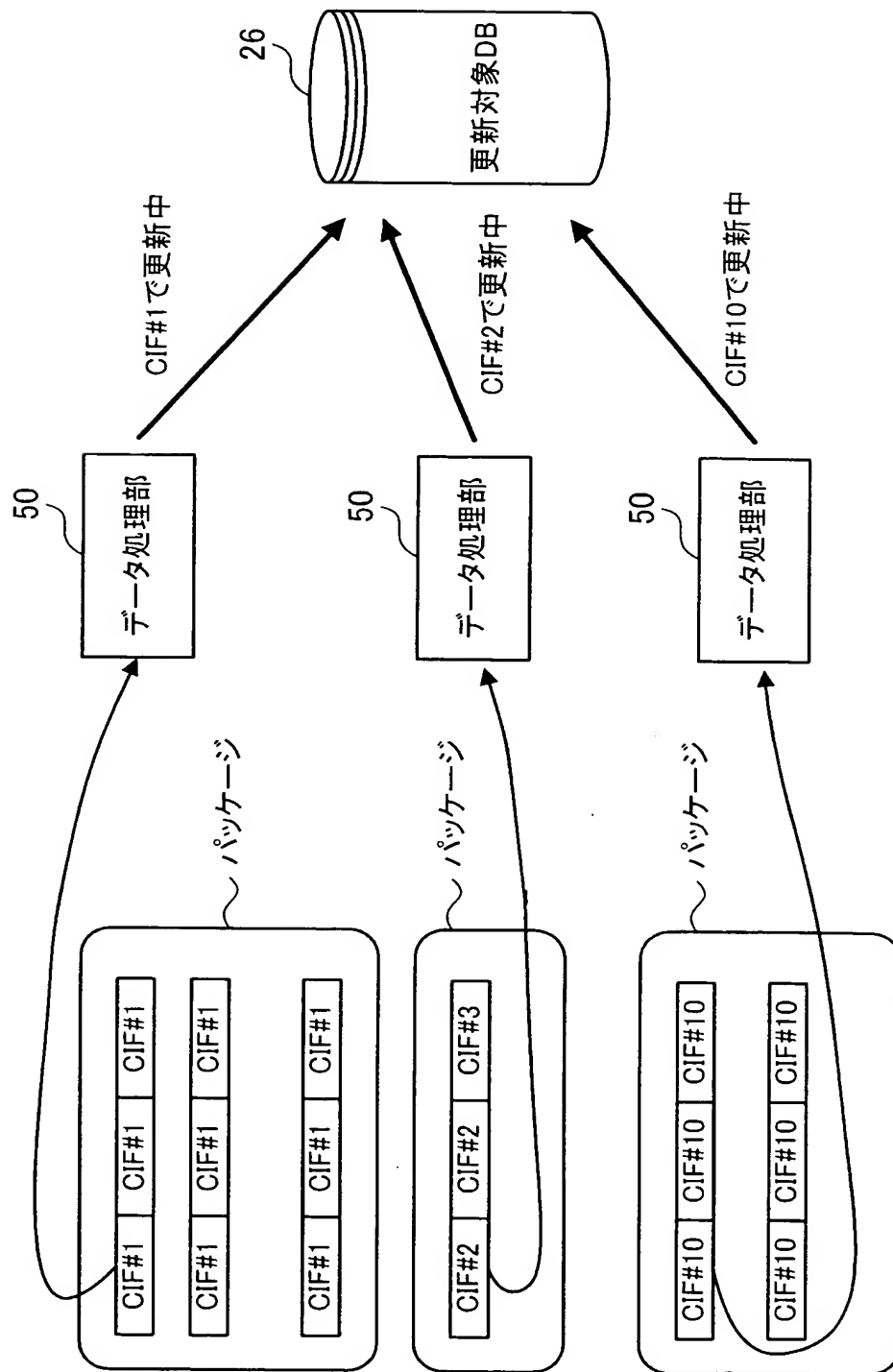
【図 1】



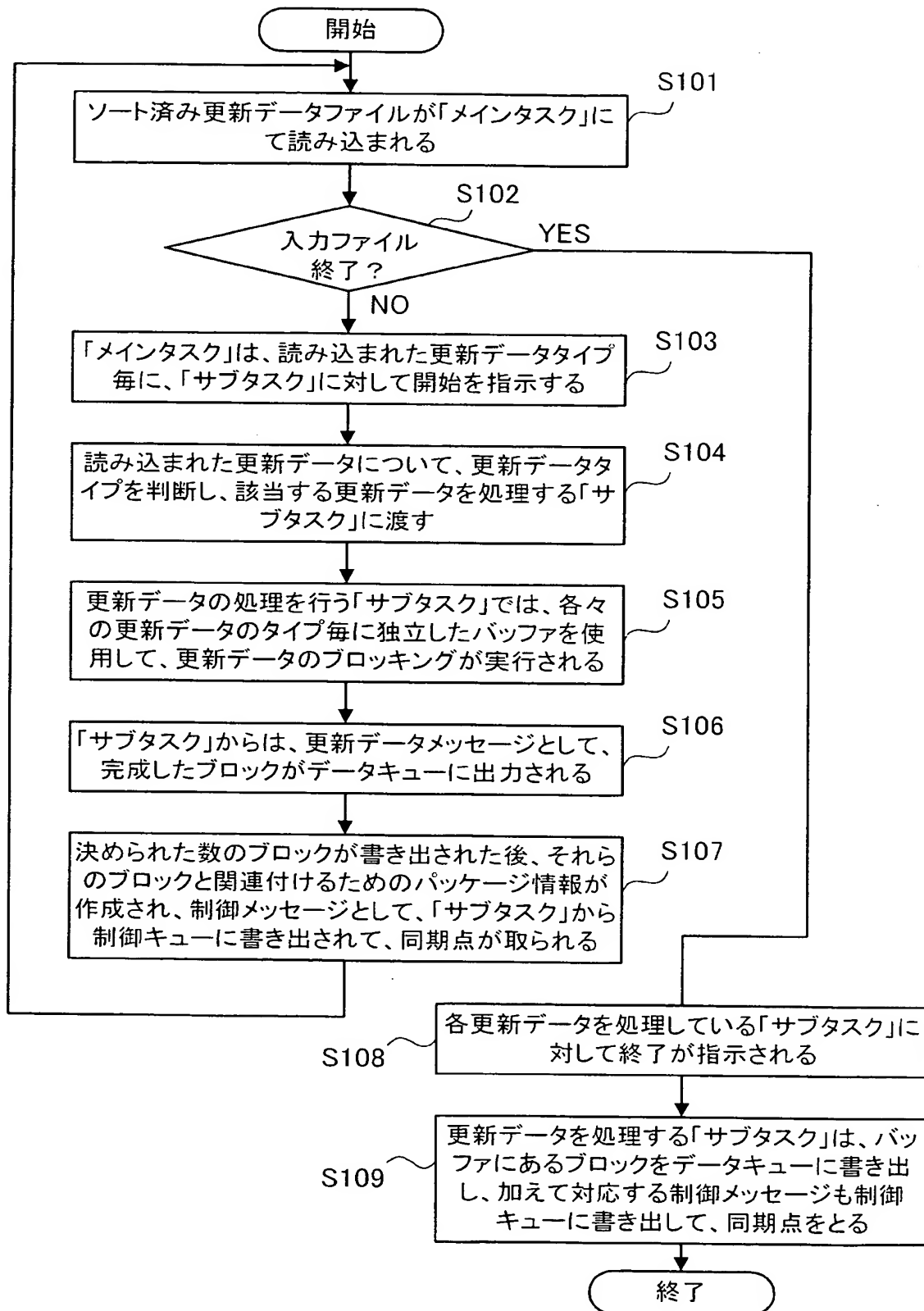
【図 2】



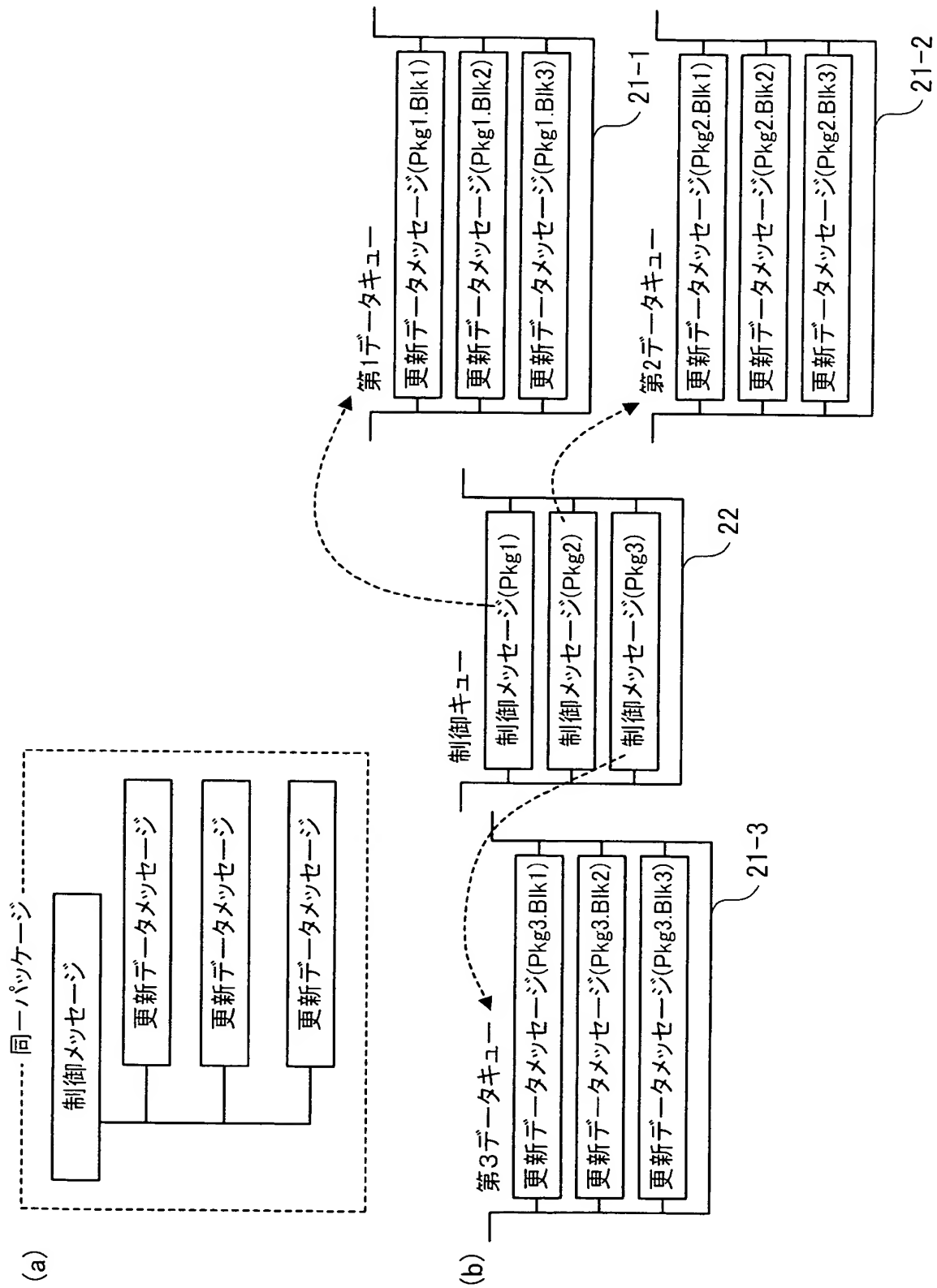
【図3】



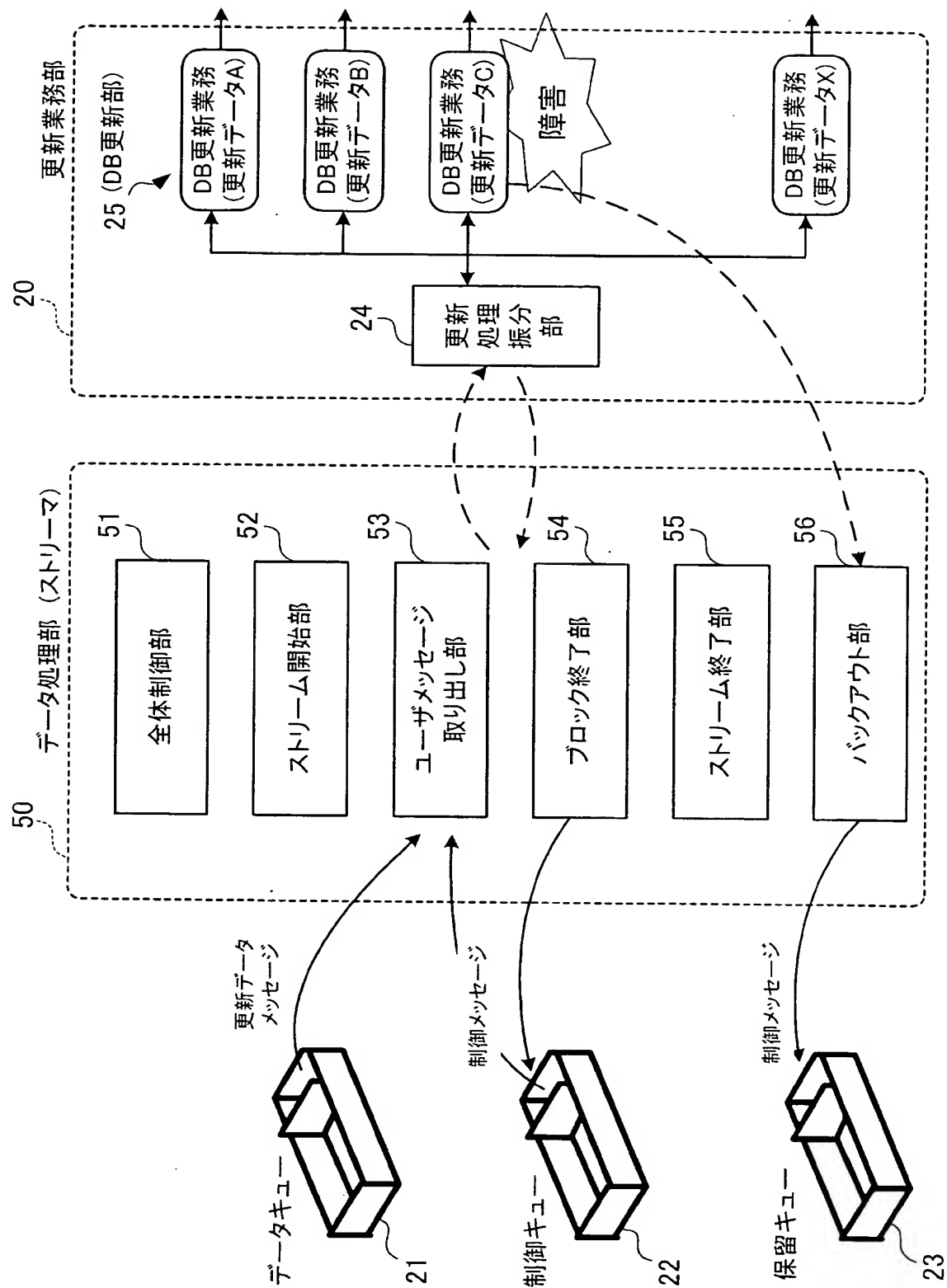
【図 4】



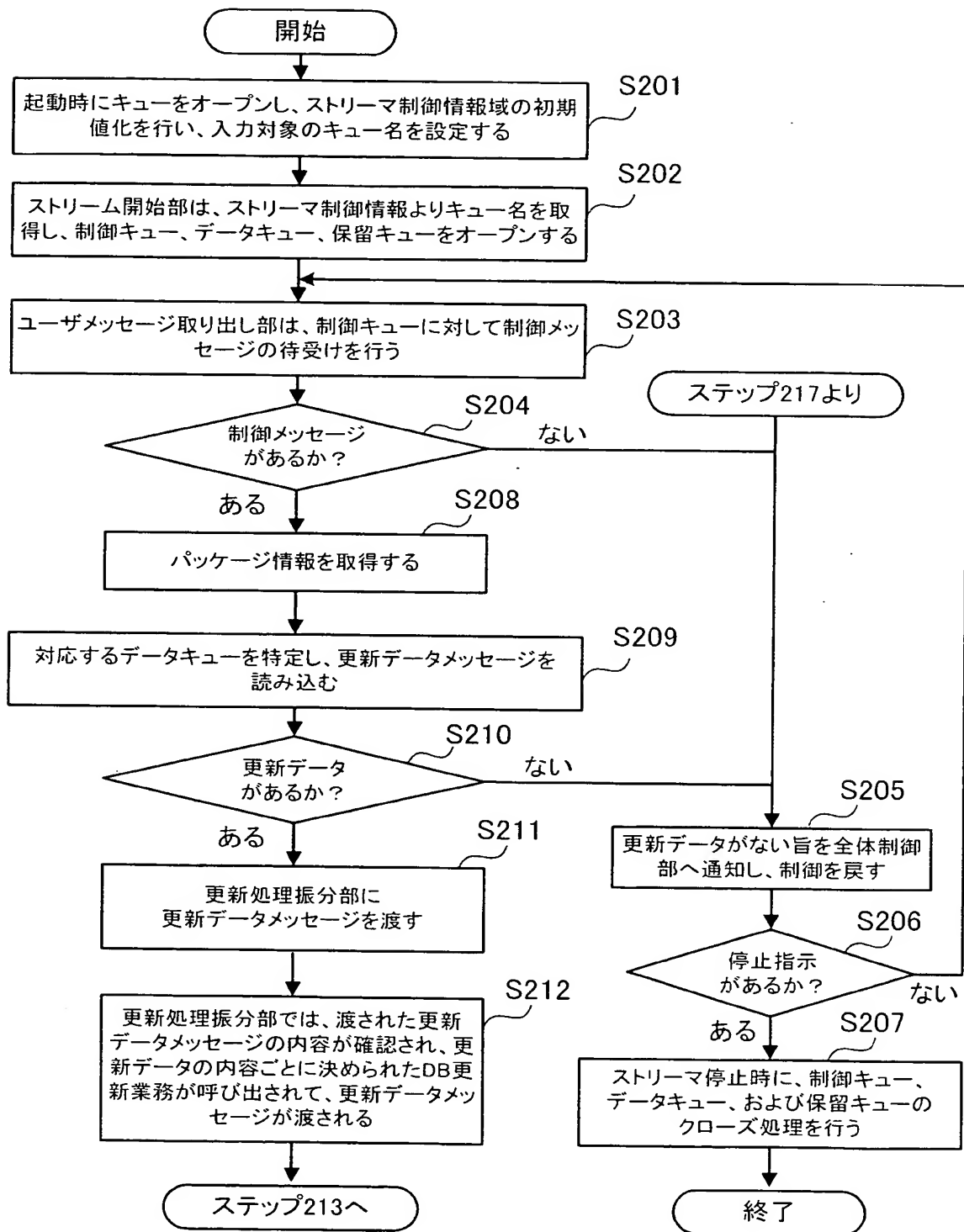
【図 5】



【図 6】

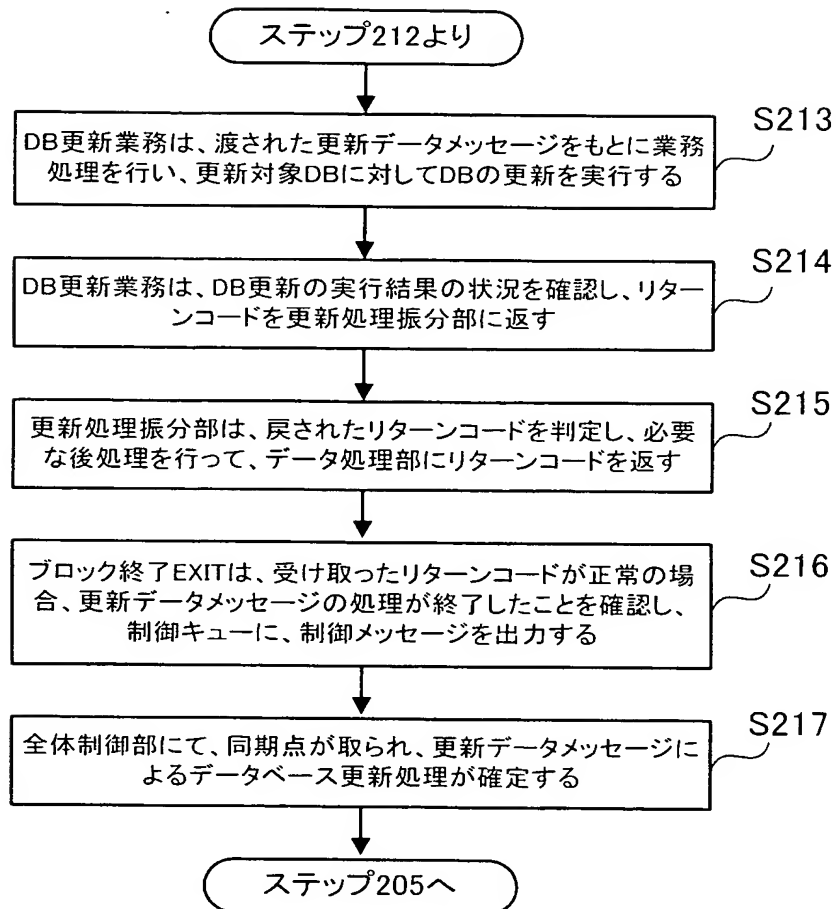


【図 7】

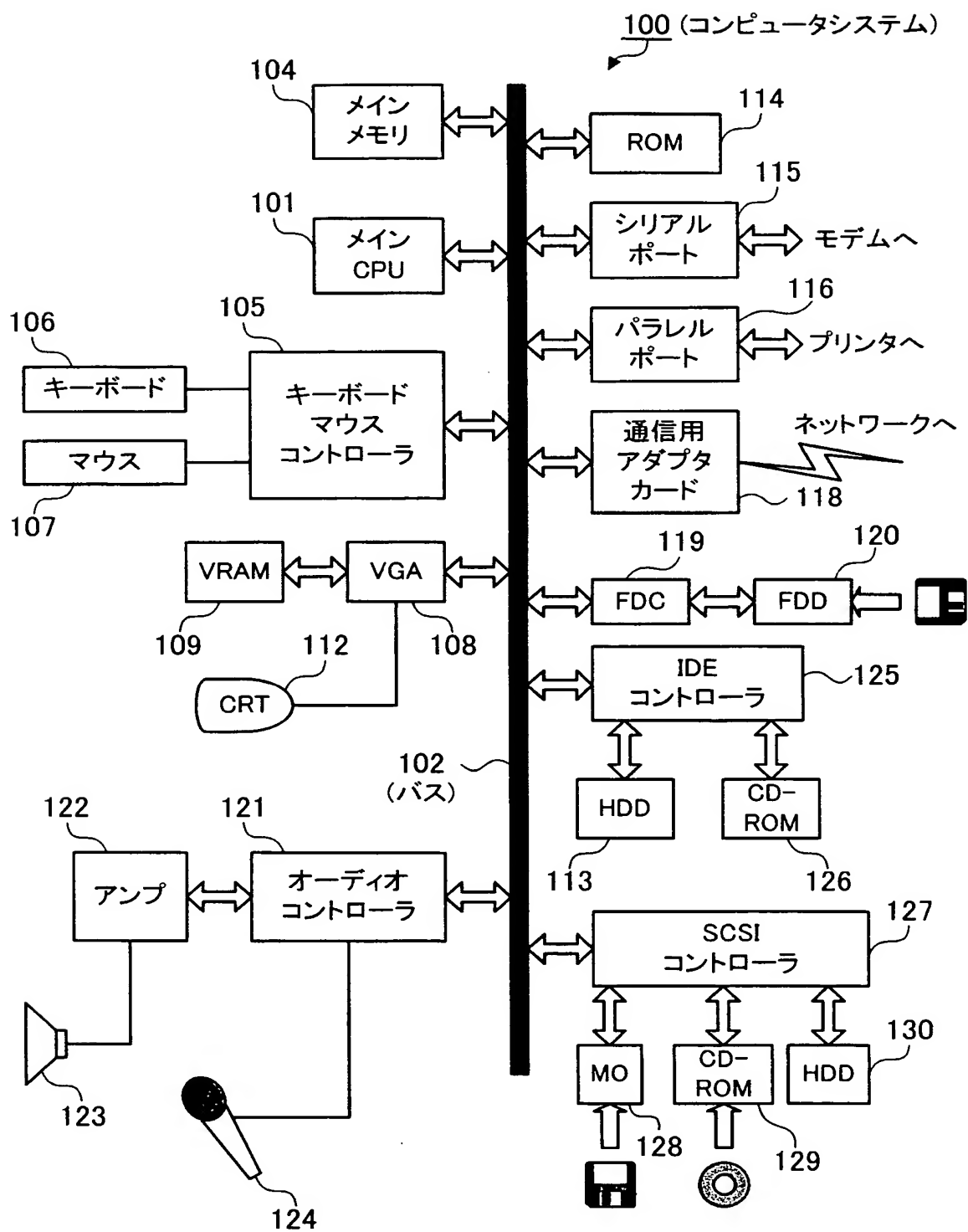




【図 8】



【図 9】



【書類名】 要約書

【要約】

【課題】 例えば制御キューという新たなコンポーネント、およびデータのブロッキングとパッケージングを採用することで、大量のデータを高速に処理する並列処理の実現と共に、データベースのコンテンションを回避する。

【解決手段】 更新対象DB 26を更新する更新データをデータメッセージとして保管するデータキュー21と、このデータキュー21に格納される複数の更新データを関連付けてパッケージ単位とし、このパッケージの内容を記述した制御メッセージを保管する制御キュー22と、制御キュー22から制御メッセージを読み取り、パッケージにより関連付けられているデータメッセージをデータキュー21から取り出してデータベース更新処理を行うデータ処理部50とを備えた。

【選択図】 図1

特願 2 0 0 3 - 0 8 0 1 6 8

出 願 人 履 歴 情 報

識別番号

[ 5 9 8 0 4 9 3 2 2 ]

1 . 変更年月日

1 9 9 8 年    3 月 3 1 日

[変更理由]

新規登録

住 所

東京都千代田区丸の内 2 丁目 7 番 1 号

氏 名

株式会社東京三菱銀行

特願 2 0 0 3 - 0 8 0 1 6 8

出 願 人 履 歴 情 報

識別番号

[ 5 9 2 0 7 3 1 0 1 ]

1. 変更年月日

1 9 9 2 年 4 月 3 日

[変更理由]

新規登録

住 所

東京都港区六本木 3 丁目 2 番 1 2 号

氏 名

日本アイ・ビー・エム株式会社